Fall 2005

CEG 434/634: Concurrent Software Design

Natsuhiko Futamura
Wright State University - Main Campus

CEG434/634
Concurrent Software Design
Fall 2005
Syllabus

Time: Monday, Wednesday, 8:00 pm to 9:15 pm

Class Room: 154RC

Instructor: Dr. Natsuhiko Futamura

Office: 335 Russ Engineering Center

Email: nfutamur@cs.wright.edu

http://www.cs.wright.edu/~nfutamur/

Phone: 775-5107

Prerequisite: CS400, CEG433/633, Operating Systems.
Expected background:
Discrete mathematics, Data structure, C or C++,
Programming experience in UNIX.

This course provides an introduction to concurrent program design in the UNIX environment. Classical problems of synchronization, concurrency , and their solutions are examined throught the course projects and through readings on operating system design.

Text books:

Required: Operating Systems Concepts 6th Ed. Silbershatz and Gavlin, Addison Wesley, 2002

Unix systems Programming: Communication, Concurrency and Threads. Robbinson and Robbins, Prentice Hall 2003

References: Interprocess Communications in UNIX: The Nooks and Crannies, 2nd Ed. Johns S. Gray, Prentice Hal 1998

Exam schedule:

Midterm: Monday, October 11, In class exam

Final exam Monday, Nov 14, 8:00-10:00PM

Programming Assignment, Homework: 25%
Mid-term: 30%
Final: 45%

Grading: The grades will be based on a midterm exam, final exam, and homework assignments. Midterm carries 30%, final exam carries 45% of the total score and homework assignments carries 25% of the grade.

A - 80% or above

B - 70% - 79%

C - 60% - 69%

D - 50% - 59%

F - below 50%

The letter grades are not intended to be curved; however, I reserve the right to curve the final grades based upon the final point distribution.

A missed exam counts as a 0. The grade A indicates excellence: To receive an A, you must demonstrate a thorough knowledge of the material throughout the course.

There will be no grades of incomplete given except when documented emergencies have made it unable for the student to finish the course.

Topics: The topics covered in the course include the following:

Process management
Process scheduling
CPU scheduling
UNIX I/O Inter-process communication
Asynchronous events
Client-Server computing
Inter-process communication and sockets
Process Synchronization (critical sections, semaphores, etc)
Threads,
Deadlocks